

# **BASCOM-AVR Debugging on MINI-MAX/AVR-C using JTAGICE mkII**

## **Quick Start Guide**

Document Revision: 1.01

Date: 27 August, 2009



**BiPOM Electronics, Inc.**

16301 Blue Ridge Road, Missouri City , Texas 77489

Telephone: 1-713-283-9970. Fax: 1-281-416-2806

E-mail: [info@bipom.com](mailto:info@bipom.com)

Web: [www.bipom.com](http://www.bipom.com)

© 2009 BiPOM Electronics, Inc. All Rights Reserved.

All trademarked names in this document are the property of respective owners.

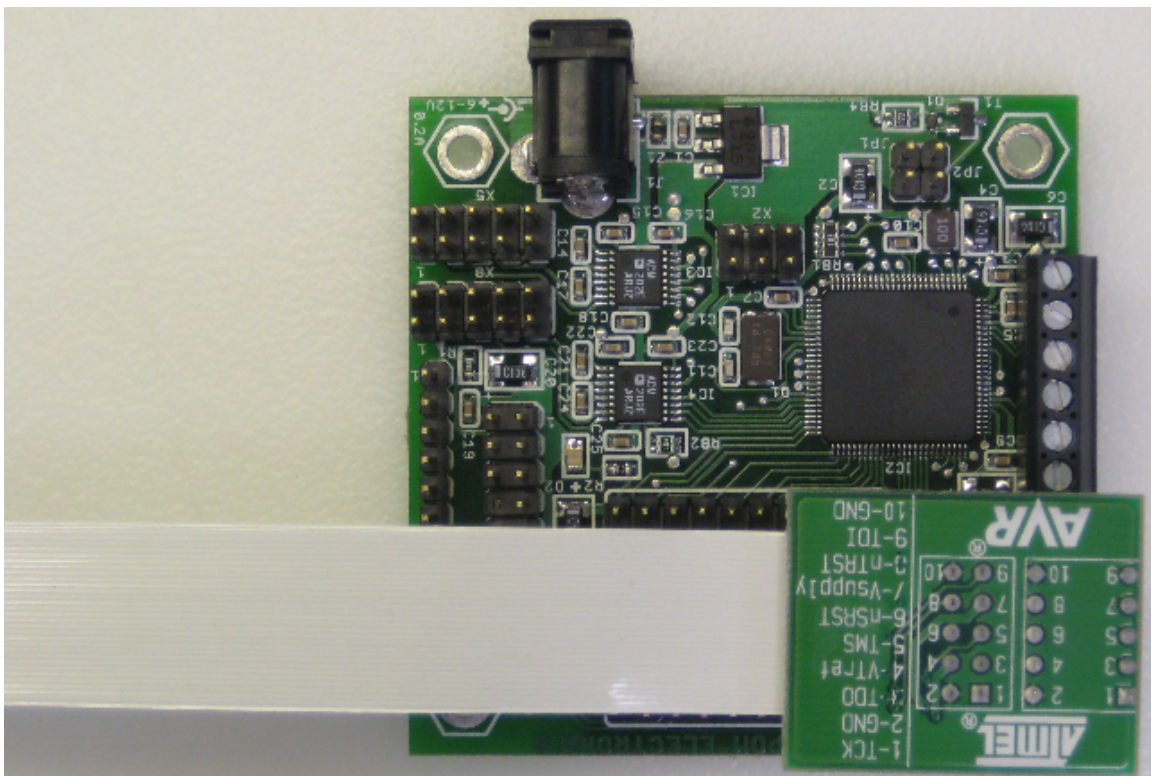
## Overview

The purpose of this document is to explain how to debug your BASIC programs written in BASCOM-AVR on the MINI-MAX/AVR-C board using ATMEL's JTAGICE mkII debugger ( also known as ATJTAGICE2 ). This debugger uses the USB port to communicate with the host PC and the JTAG port to communicate with MINI-MAX/AVR-C board.

This combination allows you to do full source level debugging of BASIC programs on the MINI-MAX/AVR-C board, including capabilities such as breakpoints, single stepping, watching variables and other features.

## Hardware Setup

Connect the JTAGICE mkII header to MINI-MAX/AVR-C's JTAG connector as shown:



JTAGICE mkII is connected to the PC through the USB port. There is no need for a separate power supply for JTAGICE mkII as it is powered from the USB port. MINI-MAX/AVR-C should be powered by its own power supply ( 6VDC adapter ).

## Software Setup

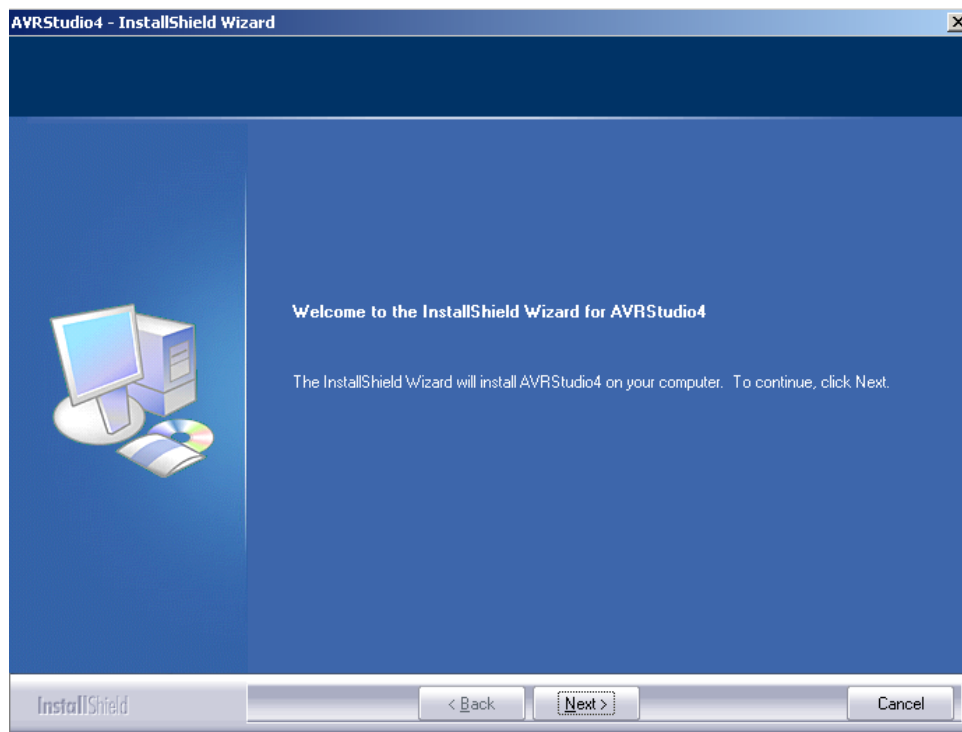
### 1. Install AVR Studio

AVR Studio® 4 from ATMEL ([www.atmel.com](http://www.atmel.com)) is a professional Integrated Development Environment (IDE) for writing and debugging AVR® applications in Windows® 9x/NT/2000/XP/Vista(32- and 64-bit) environments. AVR Studio 4 includes an assembler and a simulator.

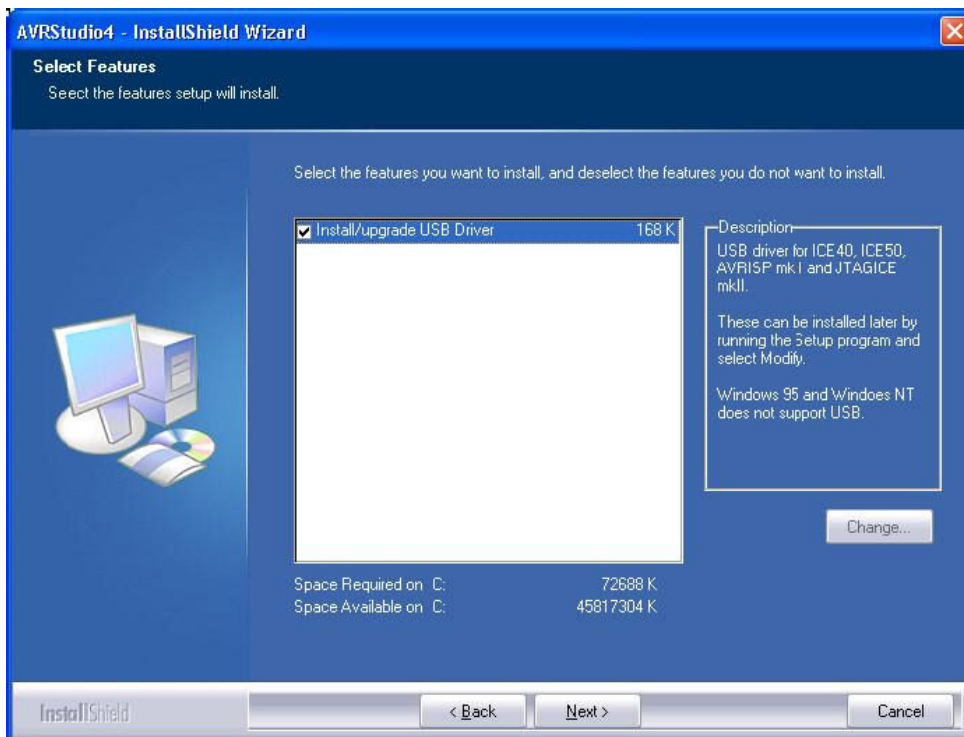
ATMEL AVR Studio is available as a free download from ATMEL website:

**Please install AVR Studio and USB driver BEFORE connecting JTAGICE mkII to your PC.**

Download and install AVR Studio 4.12 SP3 or later from <http://www.atmel.com/>



In order to use the JTAGICE mkII, it is required to install the USB driver. Check the Install/Upgrade USB Driver checkbox, and the USB Driver will be installed automatically:



When AVR Studio/USB driver installation is finished, please attach the USB cable to both your PC and JTAGICE mkII. JTAGICE mkII will be powered from the USB.

If this is the first time the JTAGICE mkII is connected to the computer, the dialog below will appear:

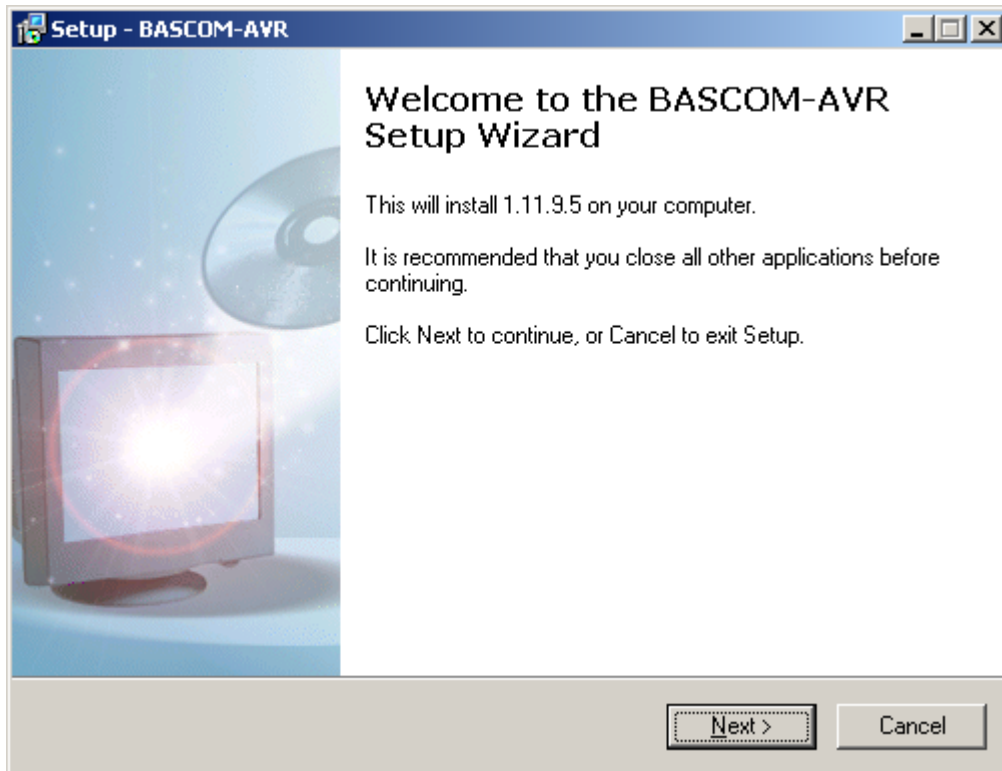




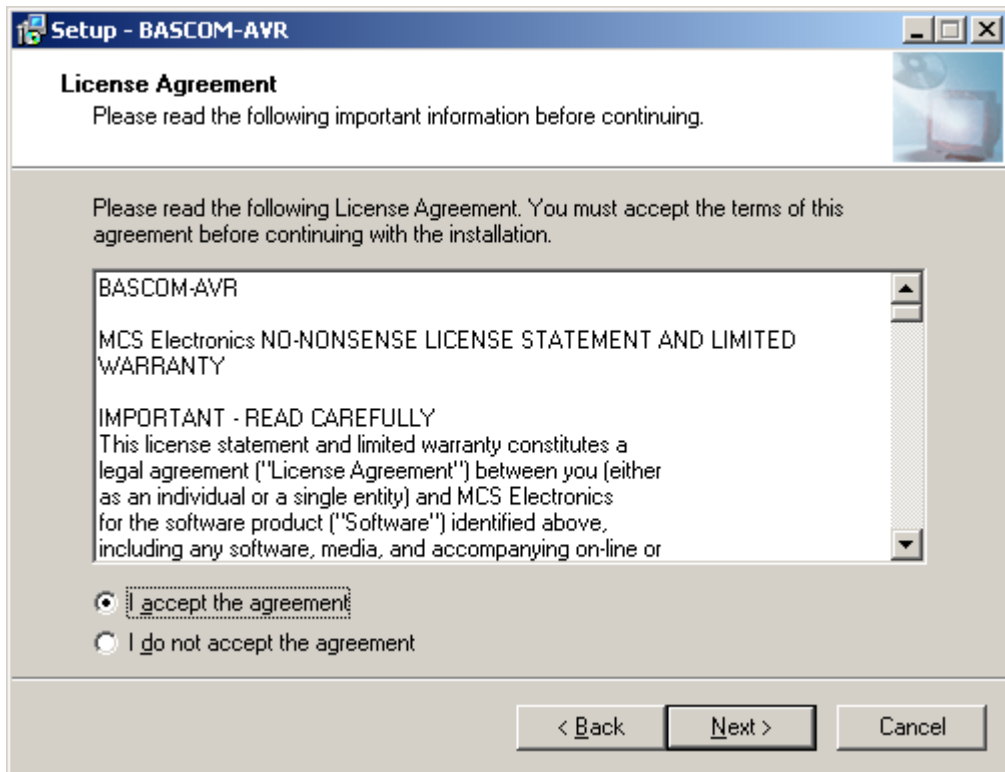
Please, click "Next" a couple of times and wait until the installation process completes. It may take from a few seconds up to a few minutes. If the AVR Studio for some reason can't detect the JTAGICE mkII, try to restart the computer in order to get the driver properly loaded.

## 2. Install BASCOM AVR

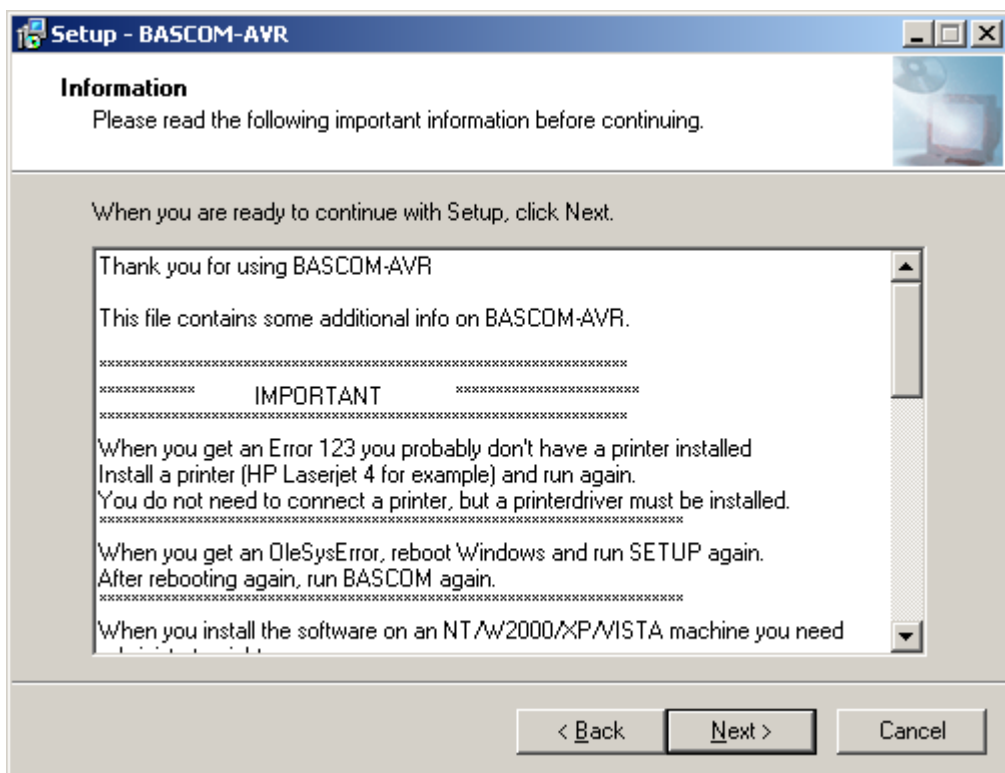
Start installation by running setup.exe:



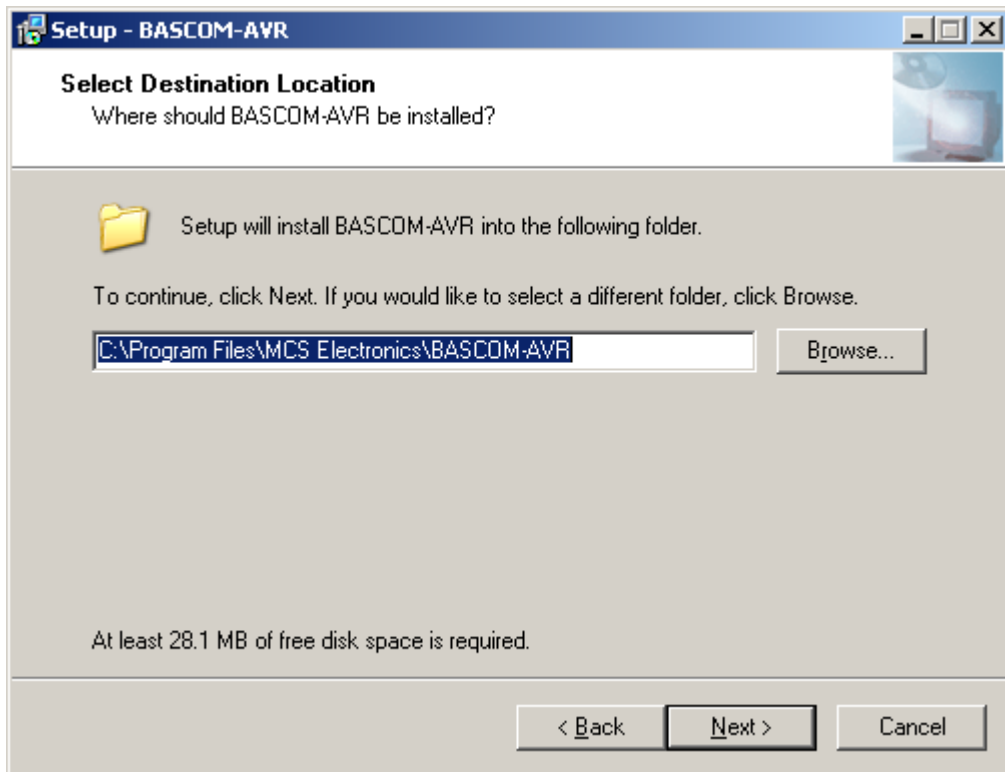
Click Next. Accept the license agreement:



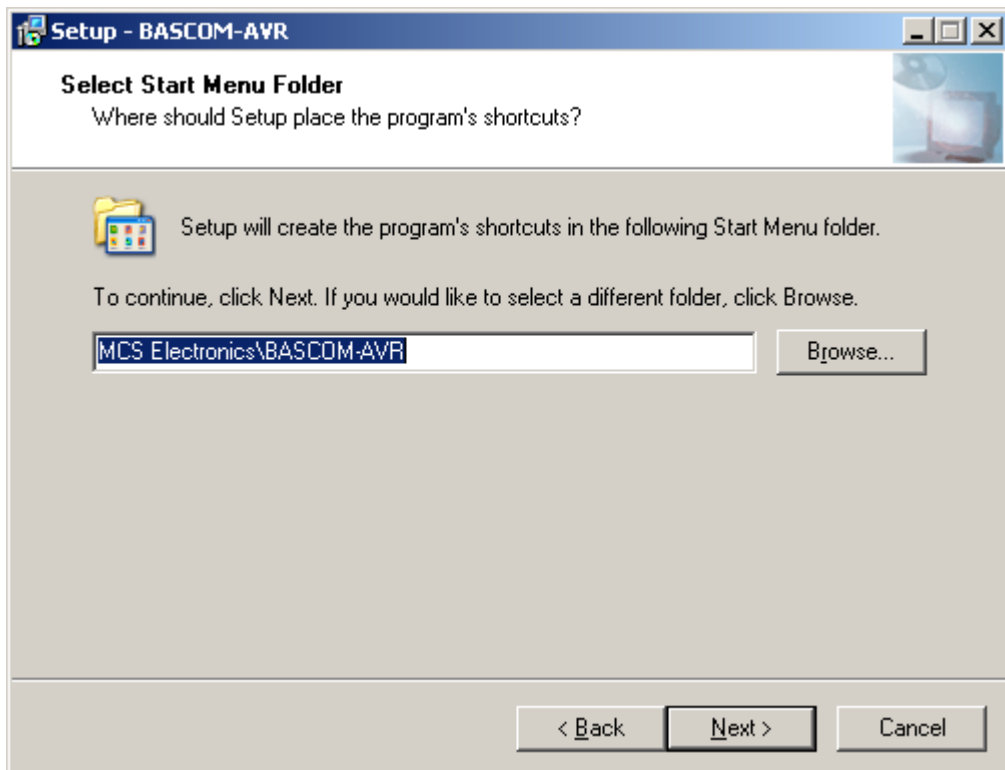
Click Next. Read the important information:



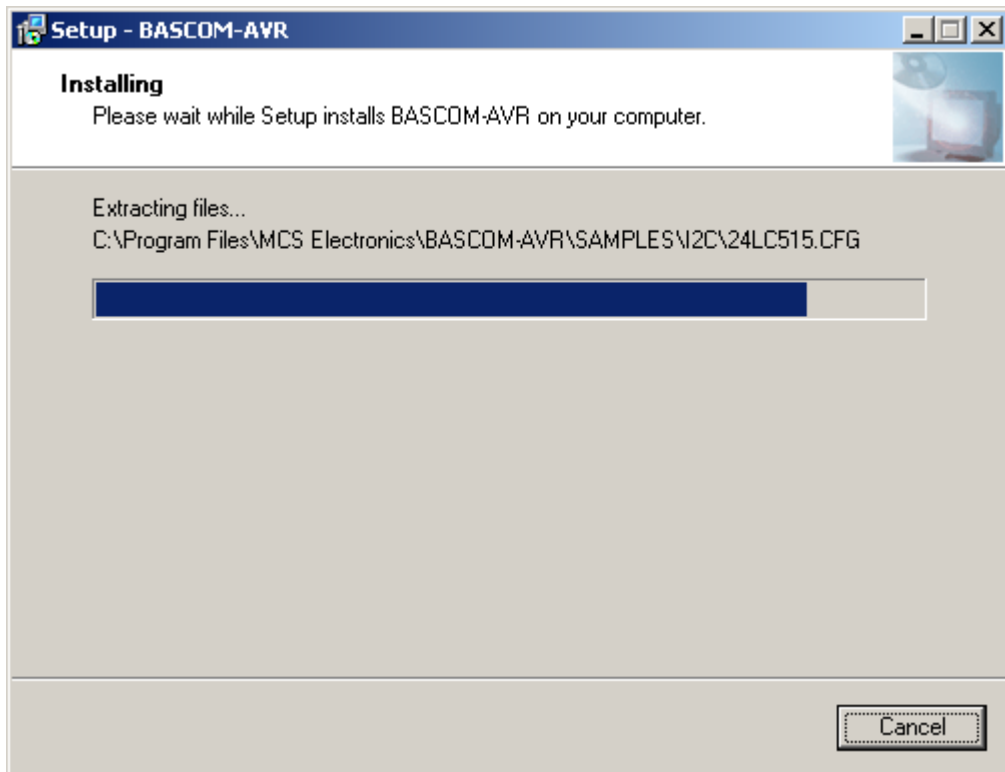
Click Next. Select the path to install. It is recommended to use the default path:



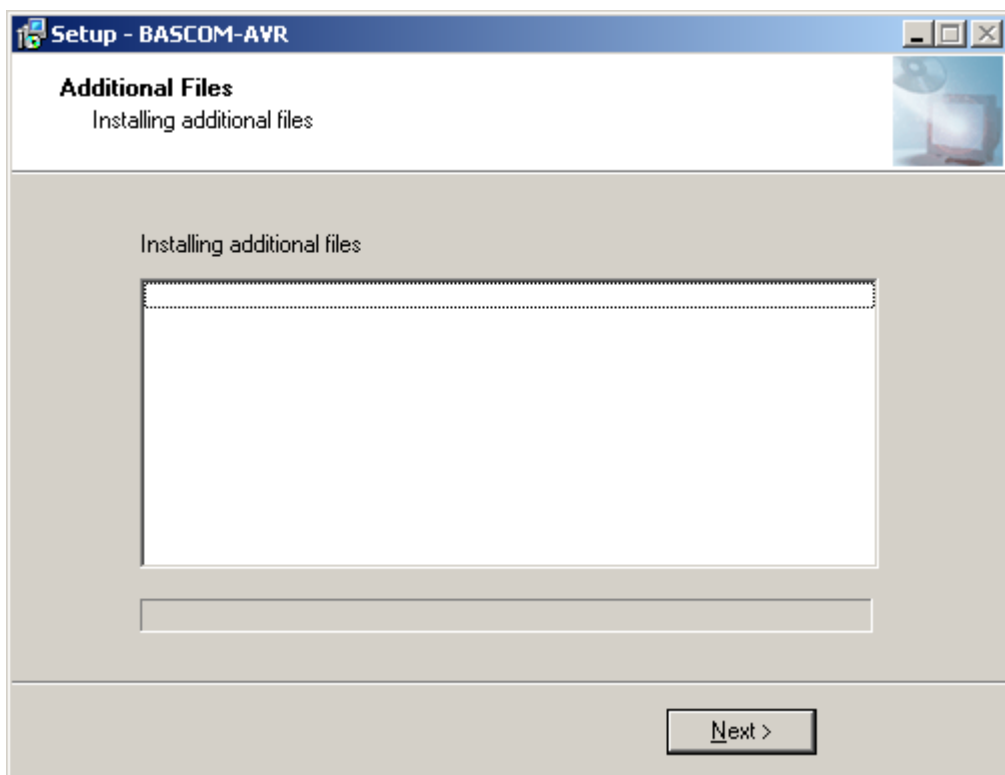
Click Next. Specify the Start Menu folder for program shortcut. It is recommended to use the default name:



Click Next. BASCOM-AVR will be installed:



Click Next. Installing Additional files dialog will appear. This may be blank. This is OK:



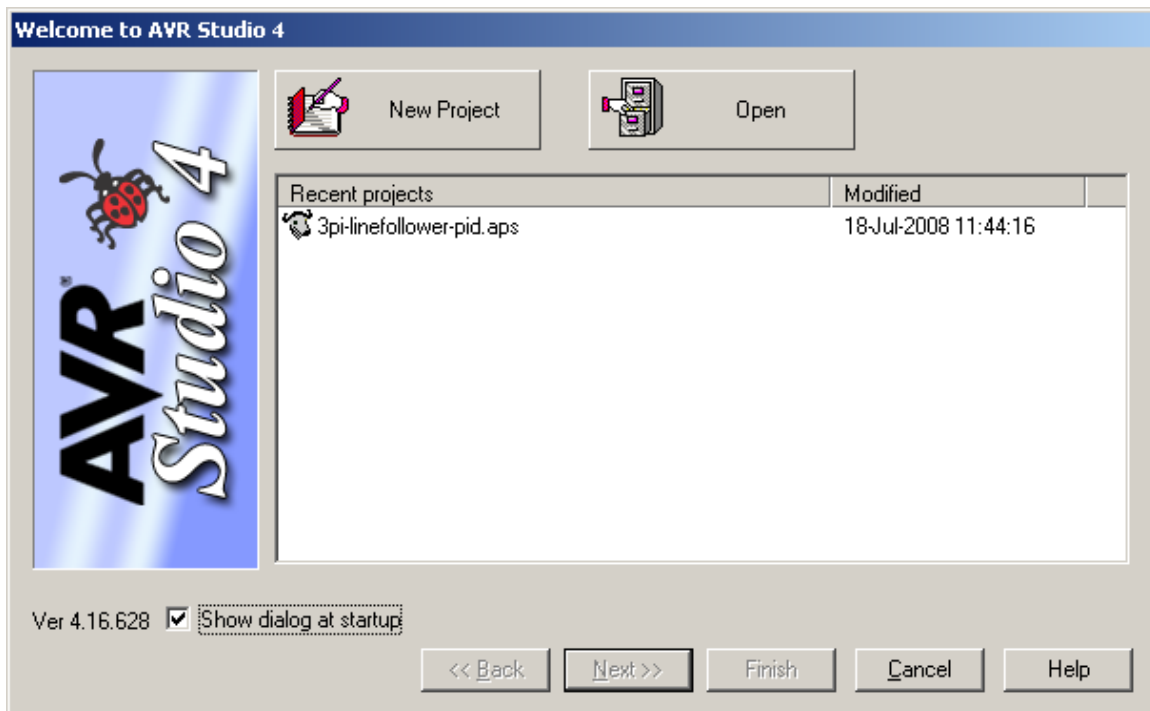
Click Next. When asked to restart your computer, select Yes and click Finish. Make sure that you have saved your data in other open applications before restarting your computer.

Follow further instructions that came with your BASCOM-AVR compiler to finish setup. For example, if you have full licensed version of BASCOM-AVR, you may need to copy the license file BSCAVR.DLL to your BASCOM-AVR folder ( default path is C:\Program Files\MCS Electronics\BASCOM-AVR )

### 3. Start AVR Studio

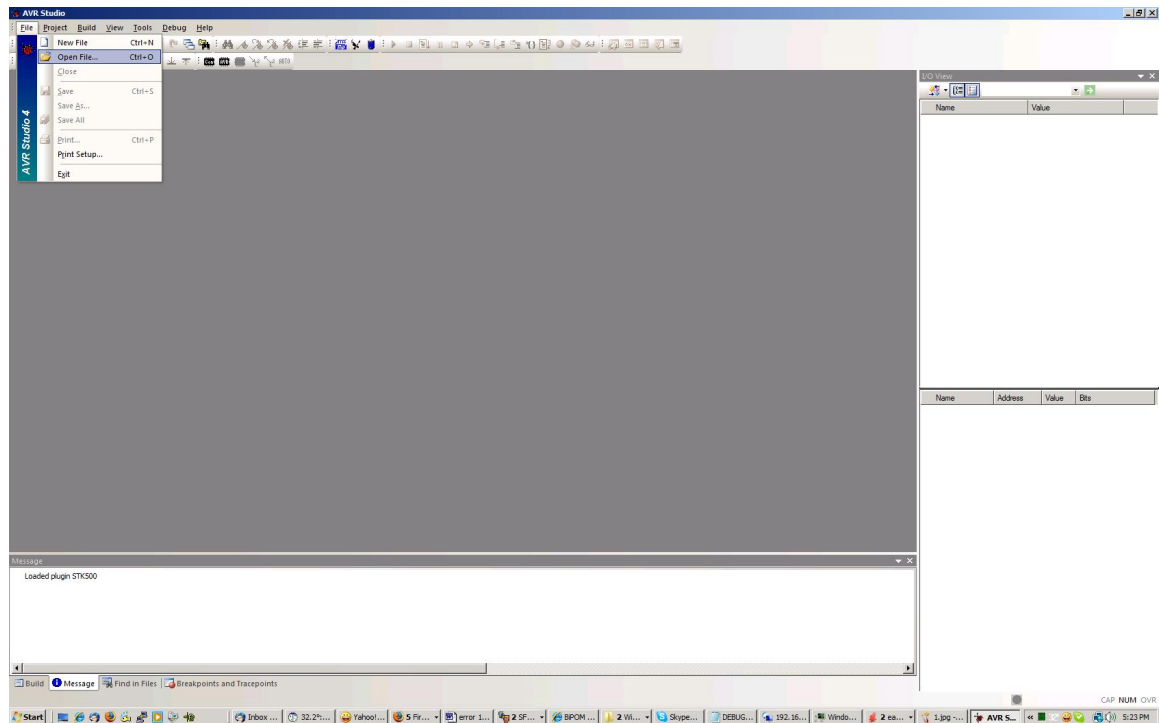


Welcome to AVR Studio dialog will appear:

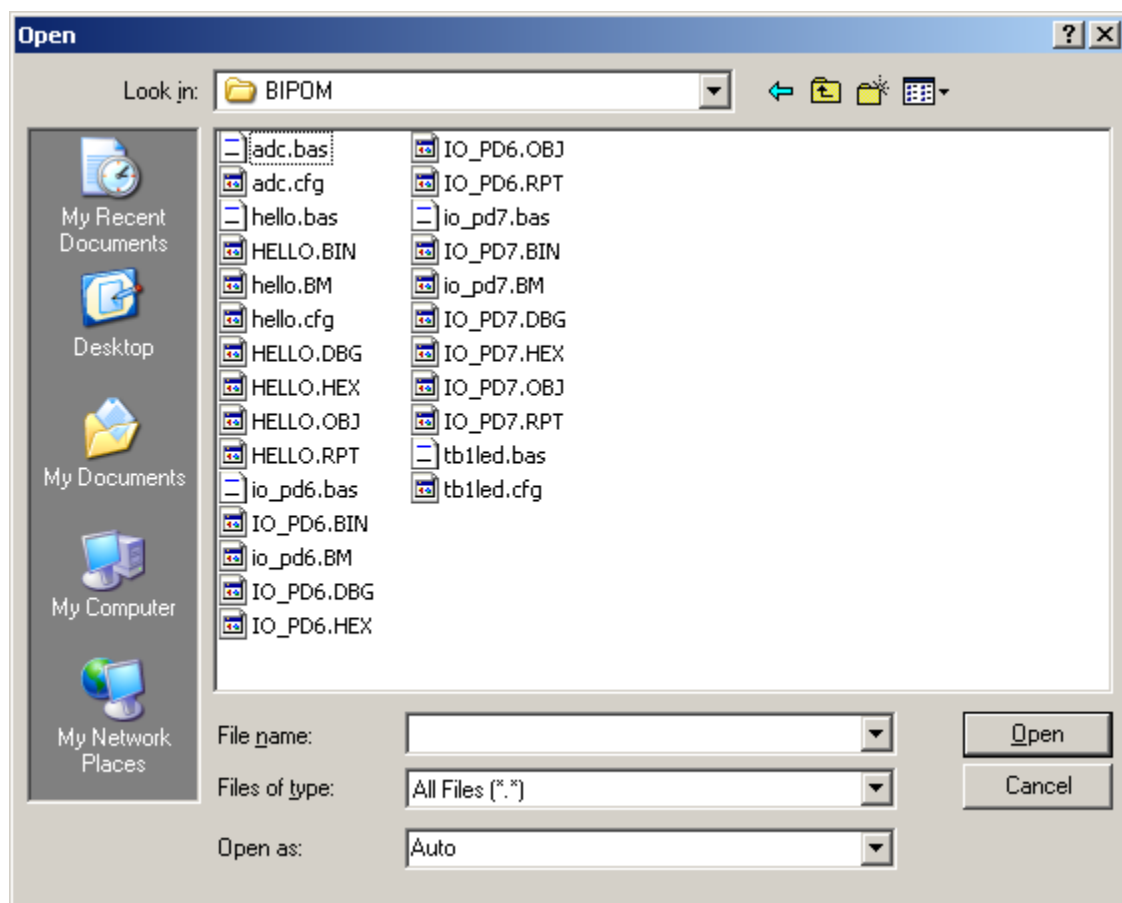


Click Cancel.

Select Open File from File menu:

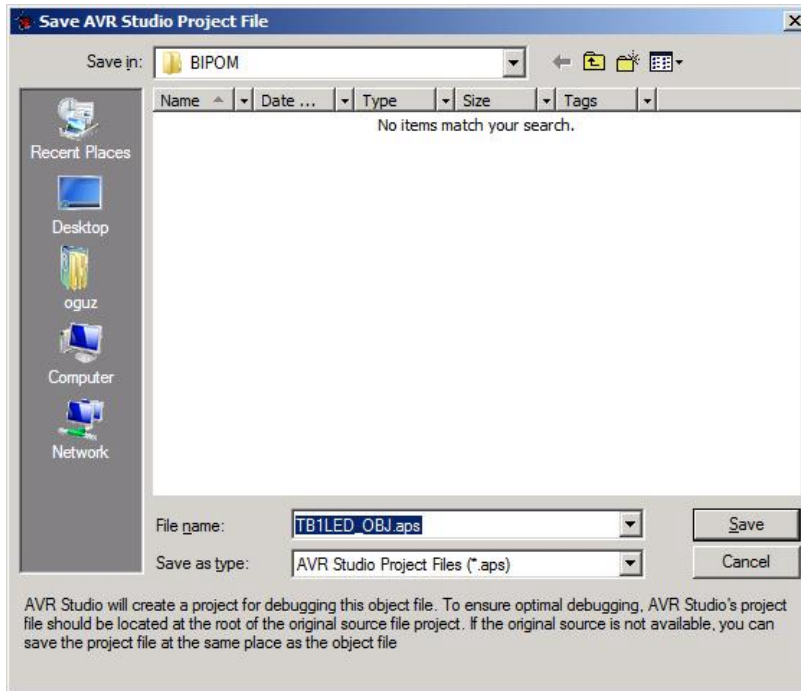


File Open dialog will appear:

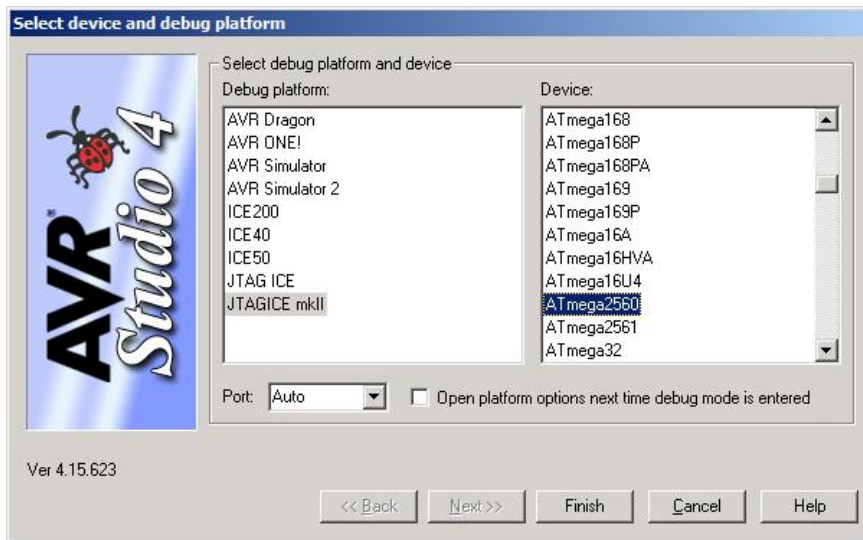




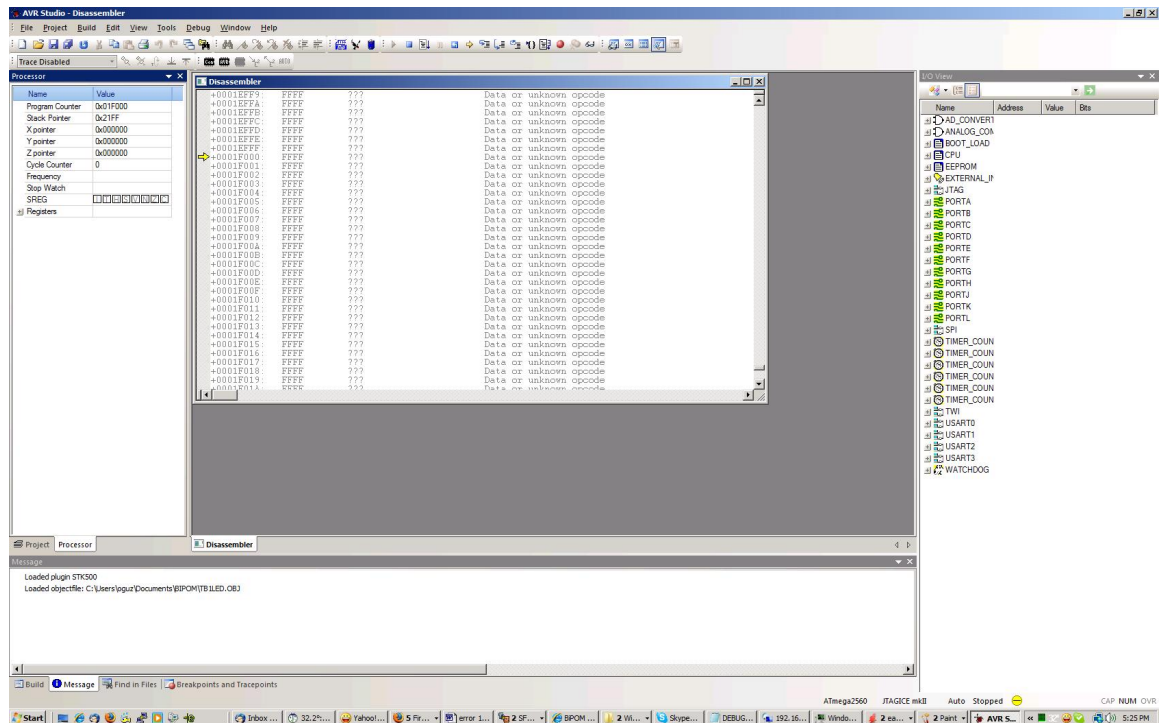
Select the Object file ( .obj ) that corresponds to the BASCOM-AVR program that you wish to debug. For example, if the BASCOM-AVR program is tb1led.bas, the object file name is tb1led.obj.



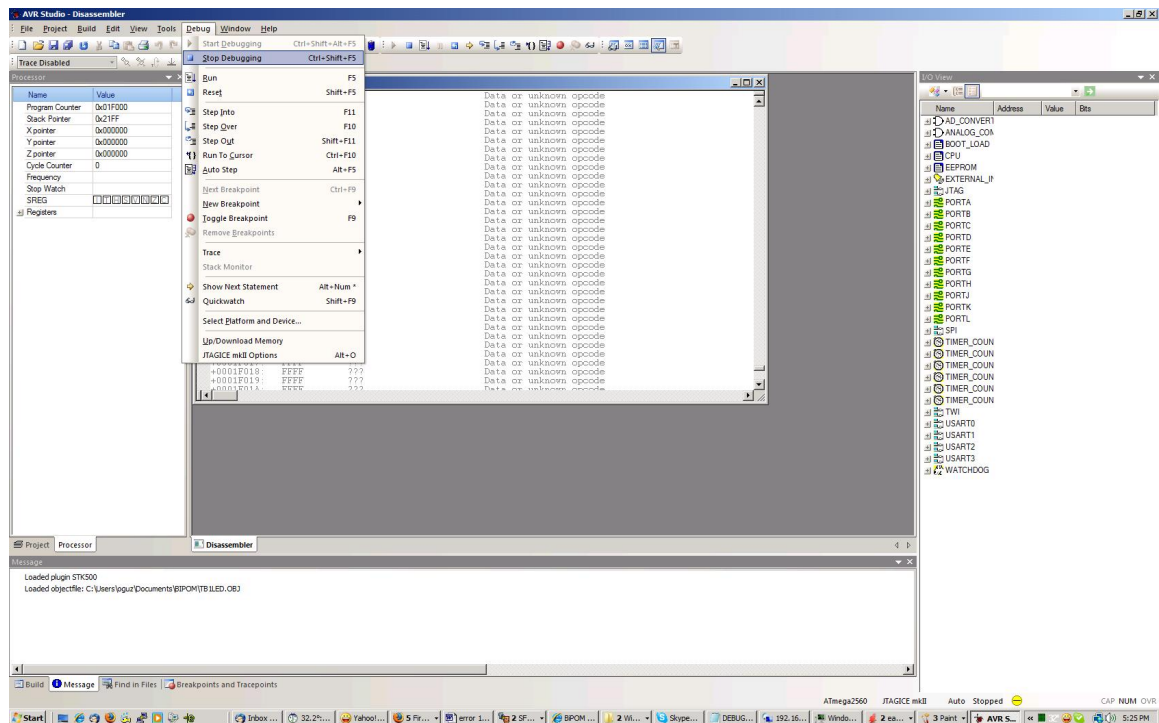
Select Device and Debug Platform dialog will appear. Select the debug platform as JTAGICE mkII and Device as ATMEGA2560. this is because the microcontroller on the MINI-MAX/AVR-C is ATMEGA2560.



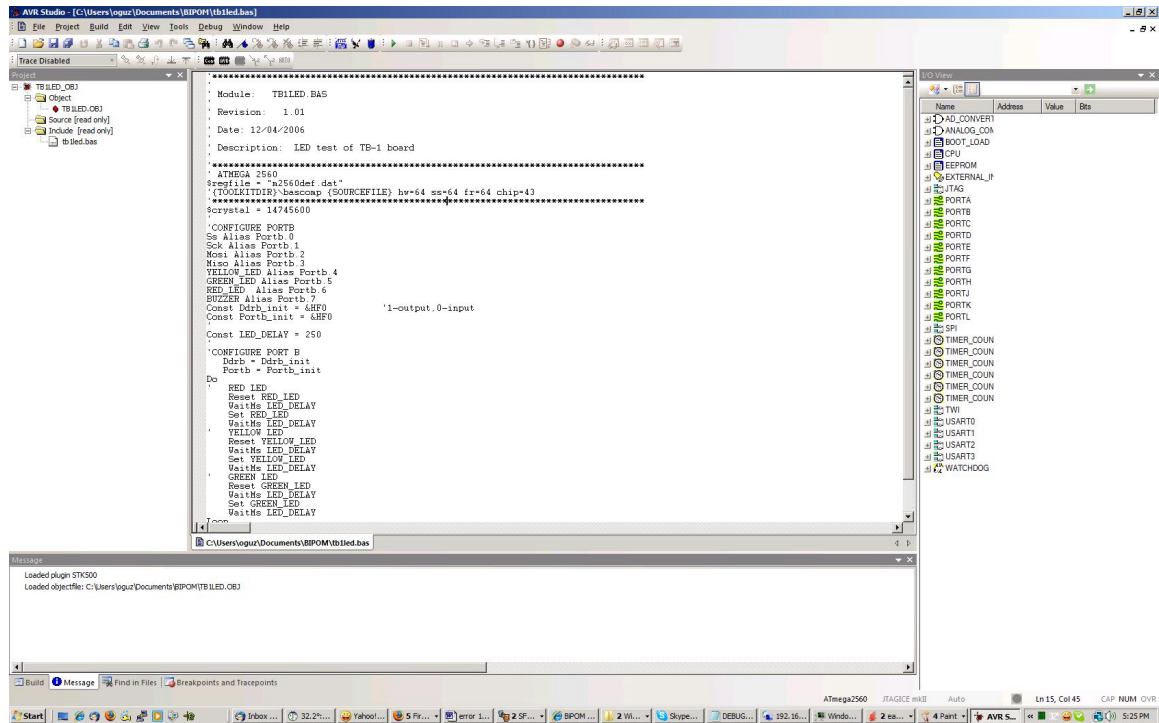
The following screen will appear. BASIC source code is not yet displayed. This is OK:



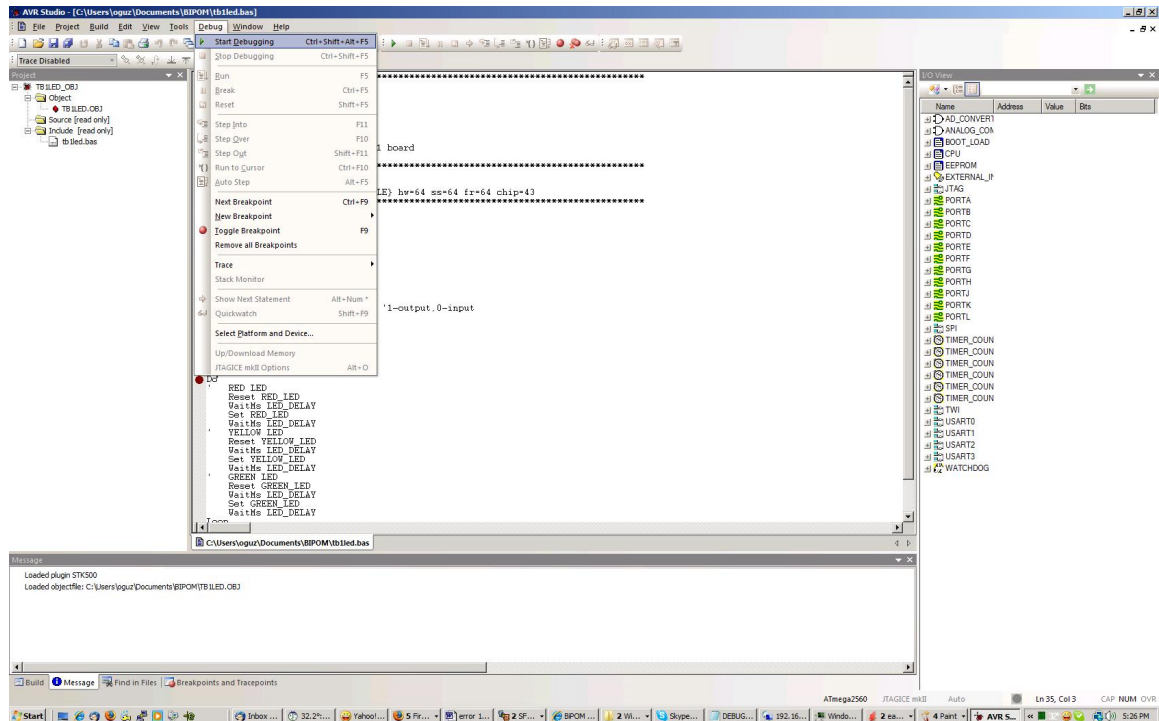
Select Stop Debugging from Debug menu:



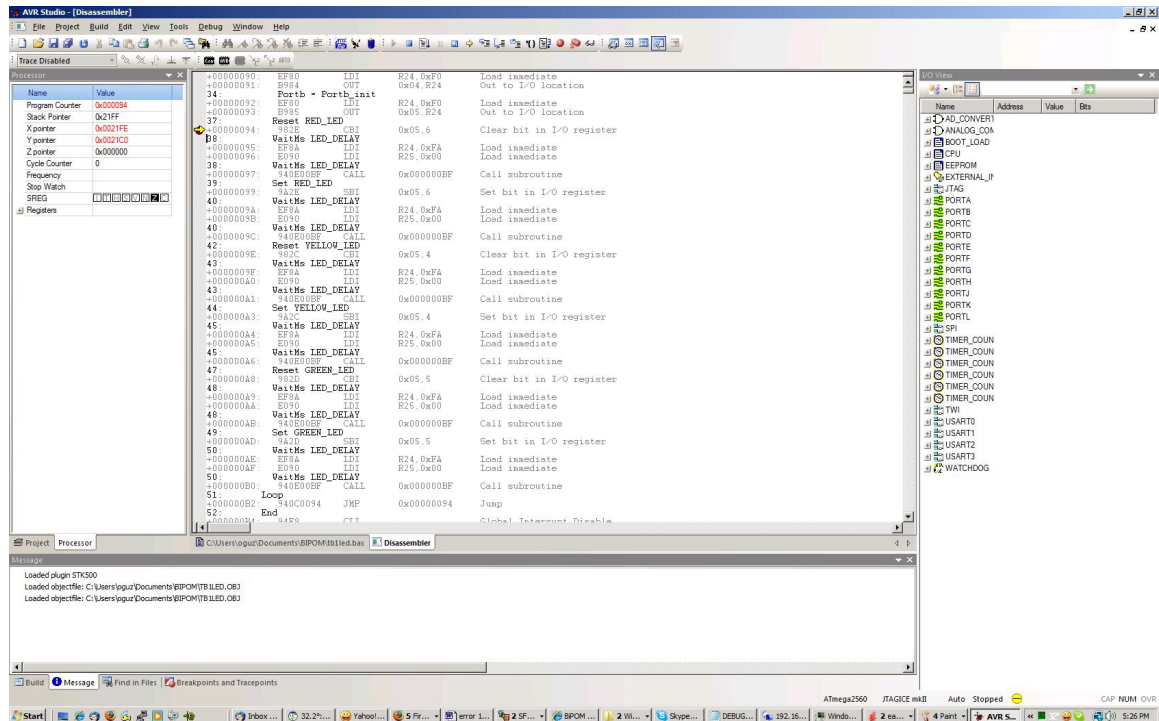
Click on the Project tab on the left most pane. The BASIC source file appears under Include folder. Double click on the BASIC source filename ( .bas ) to open it in the AVR Studio's editor ( middle pane ):



Set a breakpoint somewhere in your BASIC program by clicking on the line where you want to place breakpoint and selecting Toggle Breakpoint from the Debug menu ( F9 key does the same thing ):

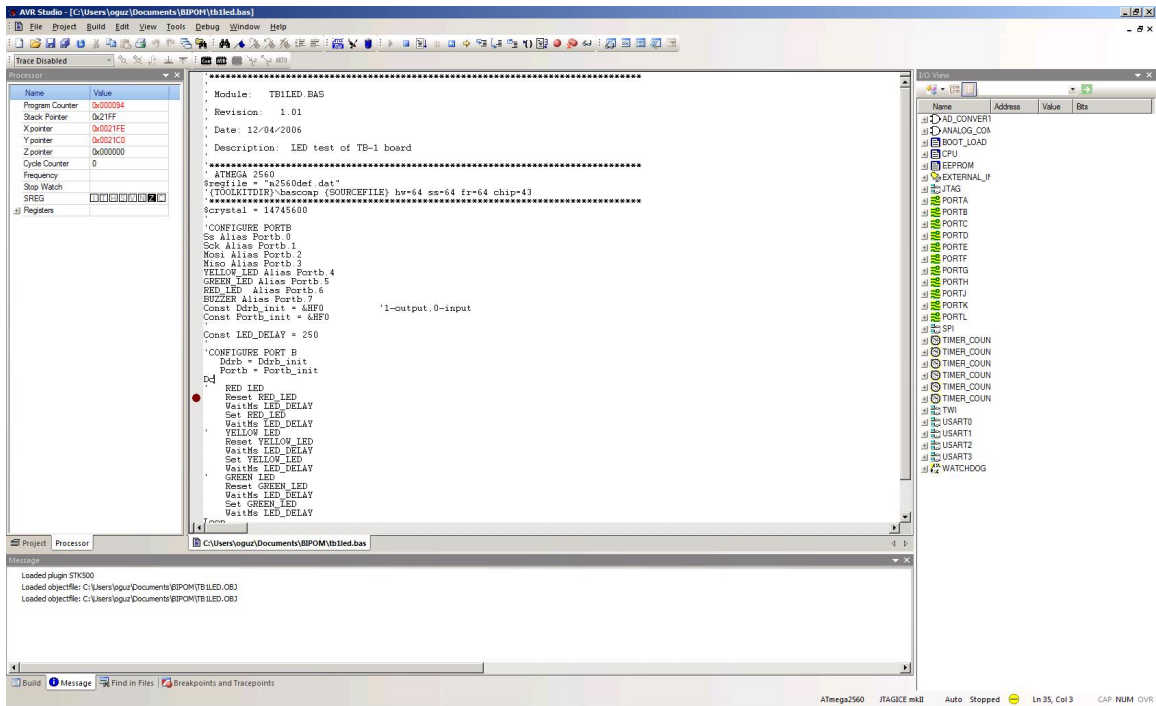


Select Run from Debug menu. The program will run until it hits the breakpoint. Program execution will stop at the breakpoint. Both the BASIC source and Assembly language equivalent will be shown:

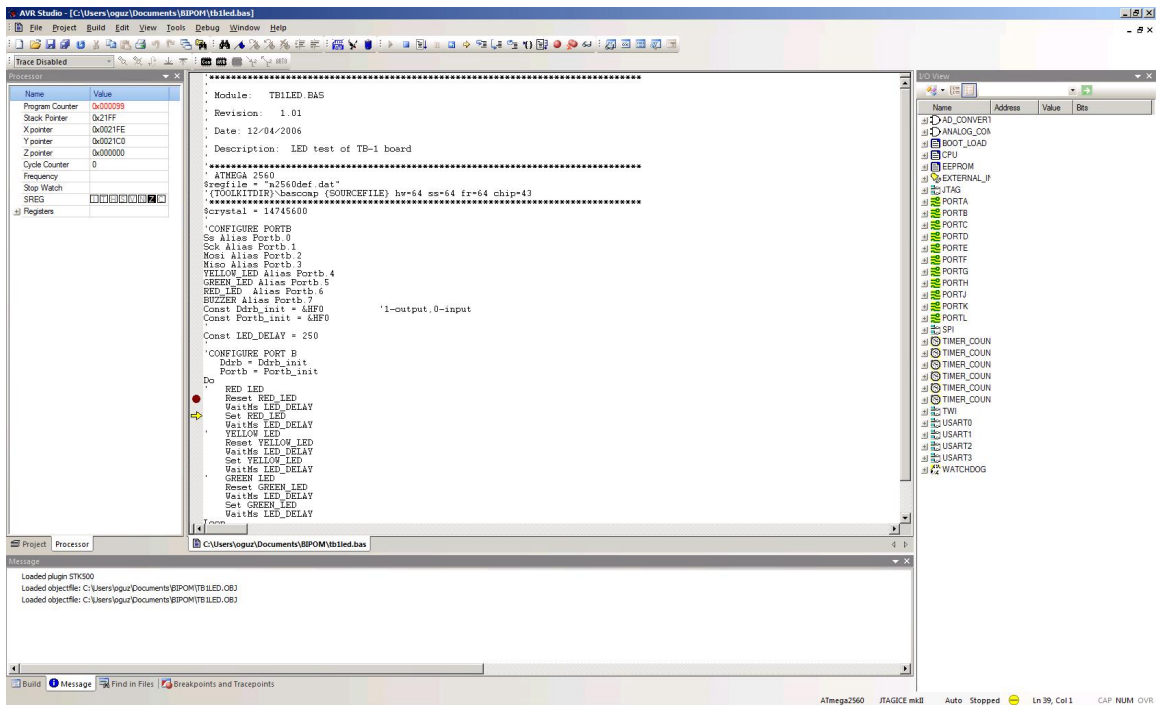


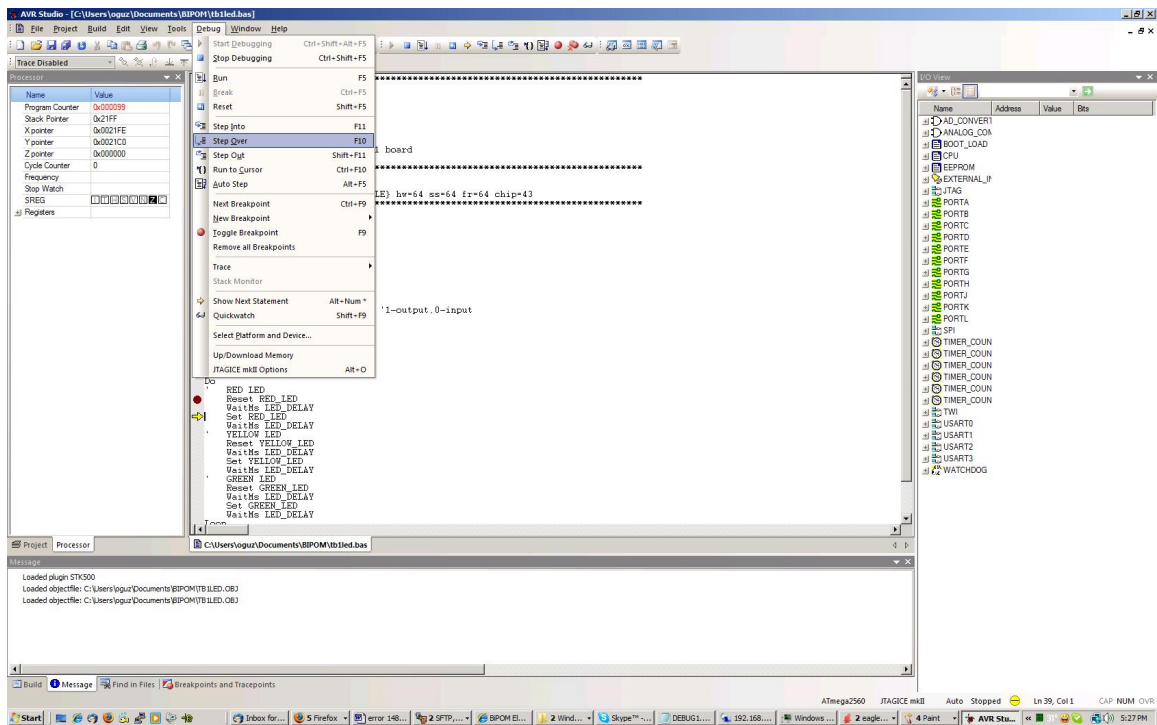


[illegible]

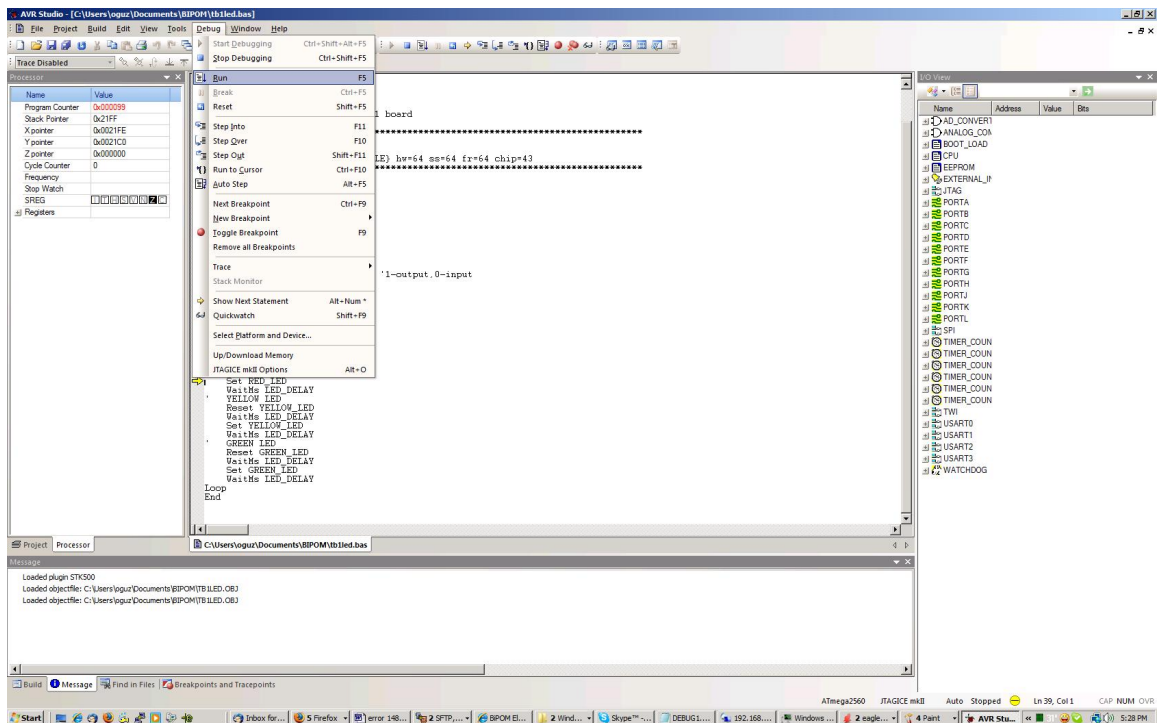


The program execution is now stopped at the breakpoint. Select Step Over under Debug menu ( F10 key does the same thing ). This will move the yellow arrow to the next code line. Yellow arrow indicates the current program position in BASIC source code.



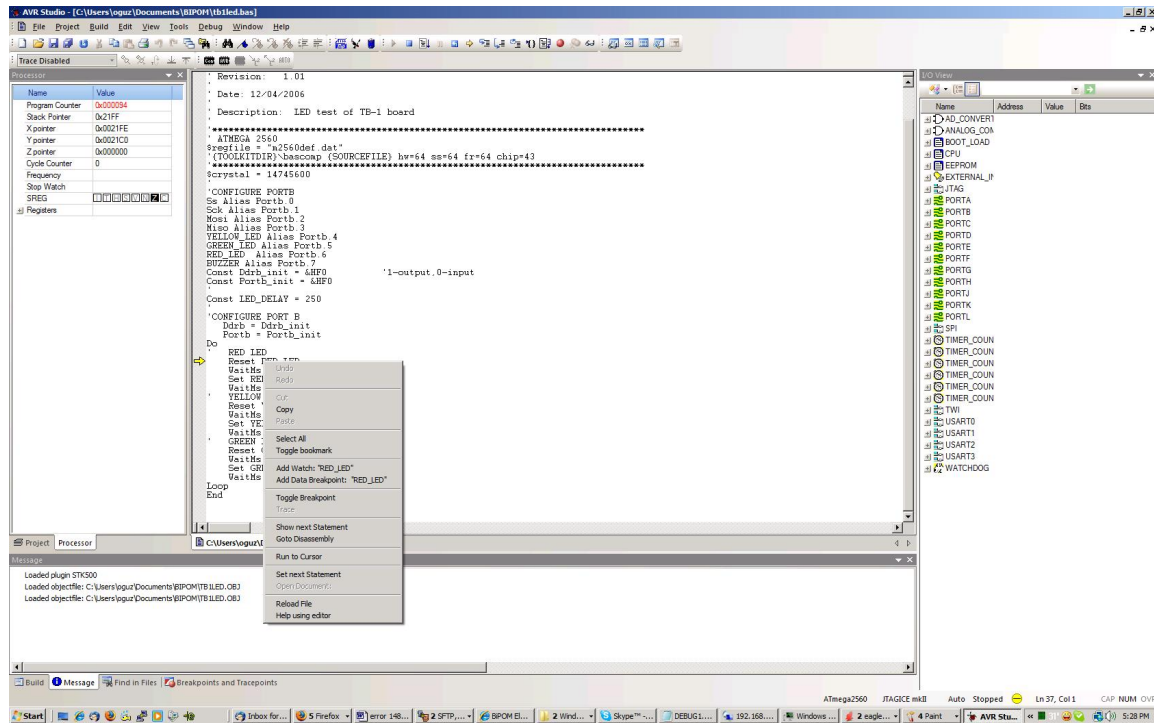


Selecting Run under Debug menu will make the program run again until it reaches the breakpoint line:

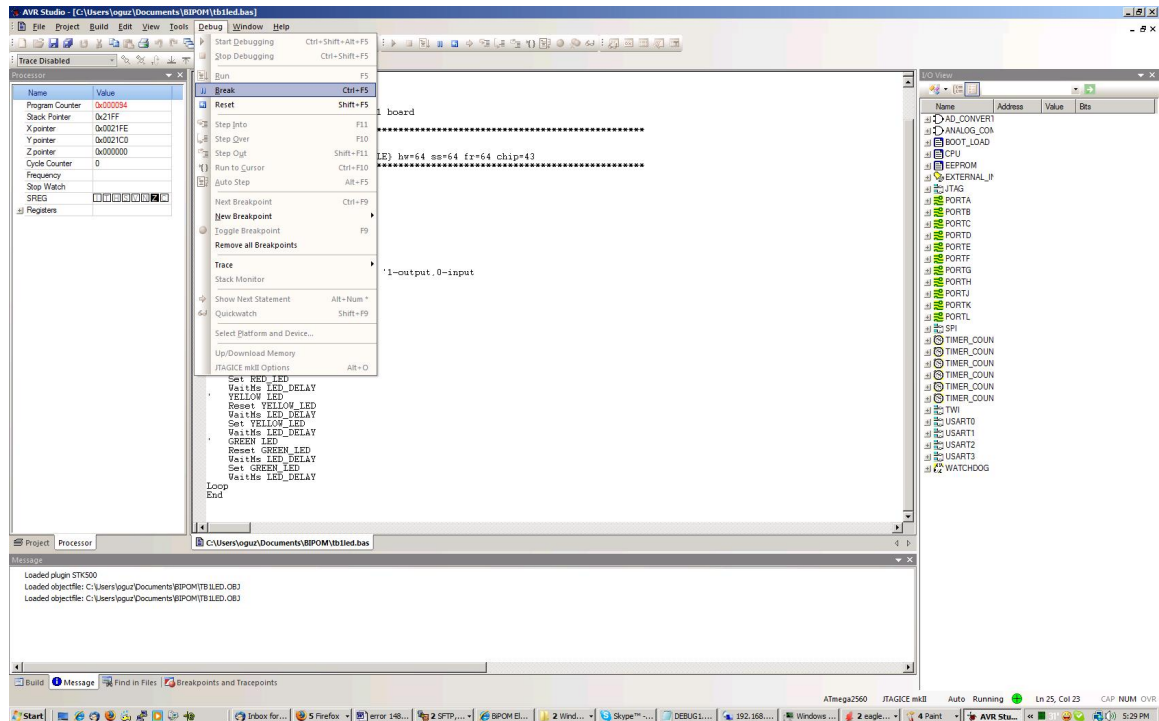


You can clear the breakpoint three different ways:

- 1) Click the mouse on the line that contains the breakpoint and press F9.
- 2) Click the mouse on the line that contains the breakpoint and select Toggle Breakpoint under Debug menu.
- 3) Click the mouse on the line that contains the breakpoint and then right click the mouse. This will bring up a menu. Select Toggle Breakpoint from that menu.



After the breakpoint has been cleared, if you select Run under Debug menu, the program will continue to run without stopping at any breakpoints ( because we cleared the only breakpoint that we added ).



To stop the program while it is executing, select Break under Debug menu. The yellow arrow in the editor will show where the program execution stopped when Break occurred.